Extending Big Data WMS to HPC and Clouds

Sergey Panitkin
Brookhaven National Lab

Next Generation “Big PanDA”

- ASCR and HEP funded project “Next Generation Workload Management and Analysis System for Big Data”
- Generalization of PanDA WMS as meta application, providing location transparency of processing and data management, for HEP and other data-intensive sciences, and a wider exascale community.
- Project participants from ANL, BNL, UT Arlington
- One of the work packages is extension of PanDA to LCF and computational clouds.

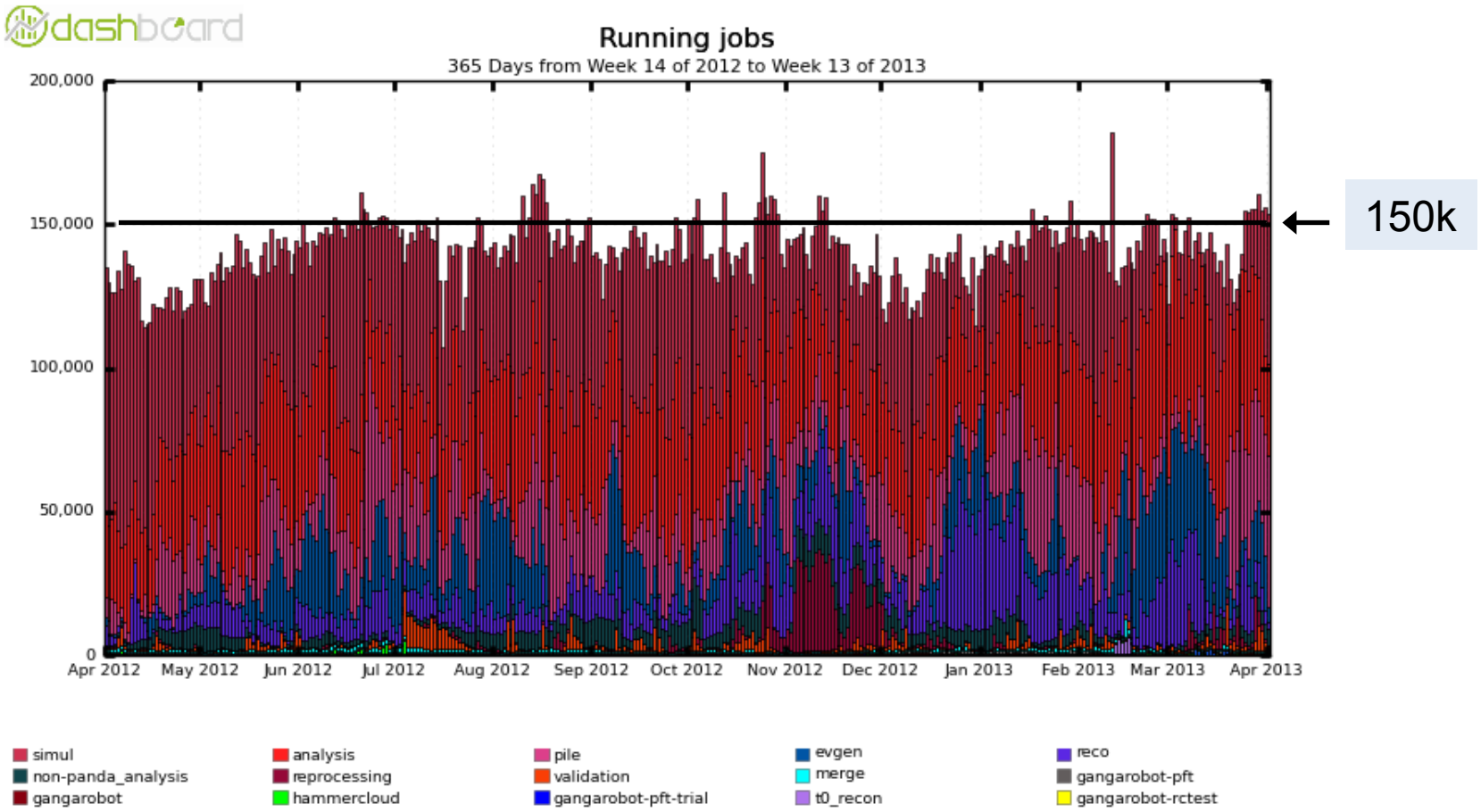
PanDA WMS

- **P**roduction **an**d **D**ata **A**nalysis (PanDA) workload management system was developed by the ATLAS Experiment at LHC
- PanDA aggregates and manages distributed collections of batch queues and storage elements that comprise LHC Grid
- It successfully manages O(10E2) sites, O(10E5) cores, O(10E8) jobs per year, O(10E3) users
- For more details see Kaushik's talk given yesterday

ATLAS - Big Data Experiment

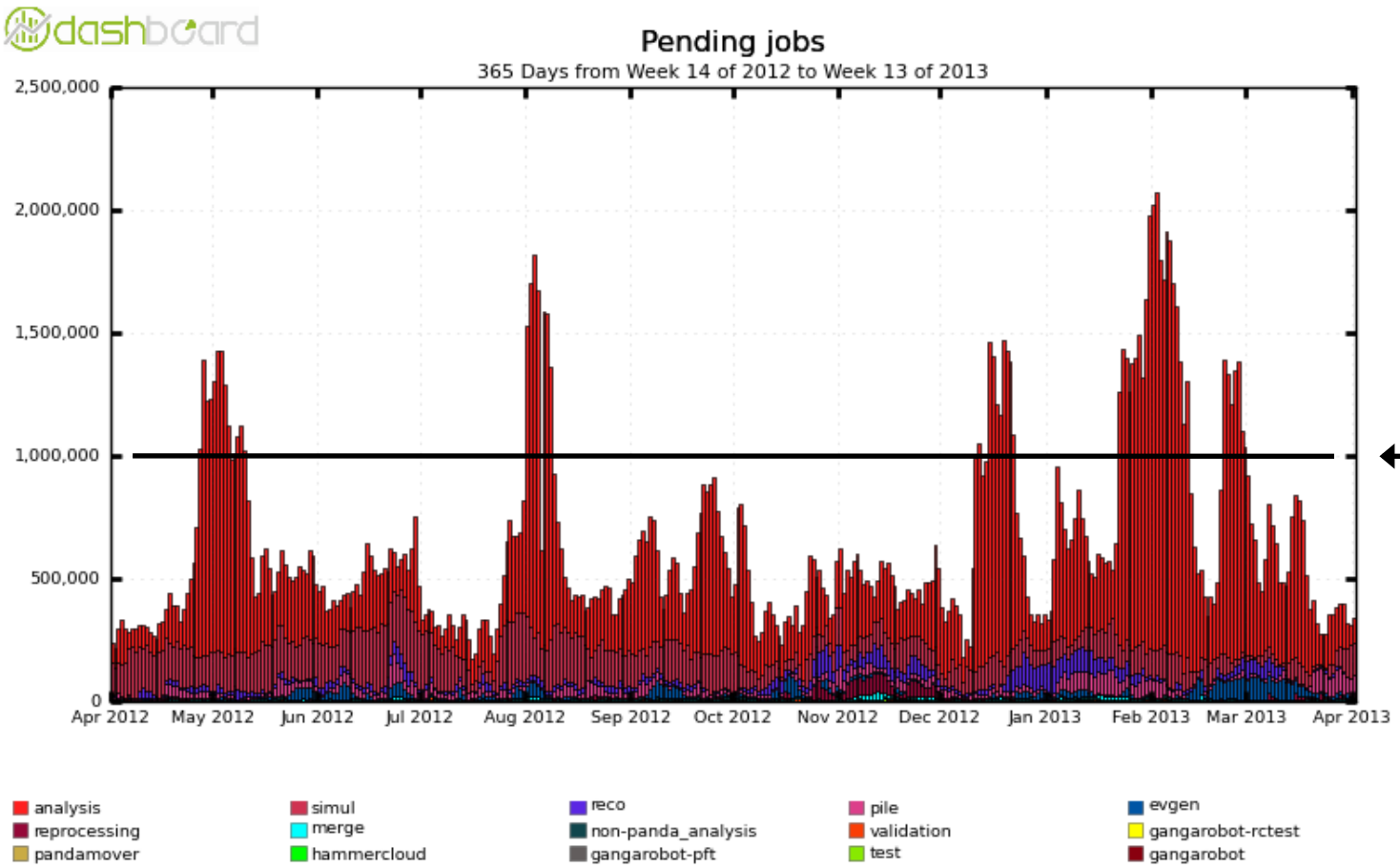
- The ATLAS experiment at the LHC - Big Data Experiment
  - ATLAS Detector generates about 1PB of raw data per second
- As of 2013 ATLAS DDM manages ~140 PB of data, distributed world-wide to ~100 of WLCG computing centers
- Expected rate of data influx into ATLAS Grid ~40 PB of data per year
- Thousands of physicists from ~40 countries analyze the data

ATLAS Distributed Computing



- Includes user and group analysis and Monte-Carlo simulations on ATLAS Grid
- Running on ~100,000 cores worldwide
- Available resources fully used/stressed

ATLAS Distributed Computing II



← 1M

Spikes in demand for computational resources
Can significantly exceed available ATLAS Grid resources
Lack of resources slows down pace of discovery

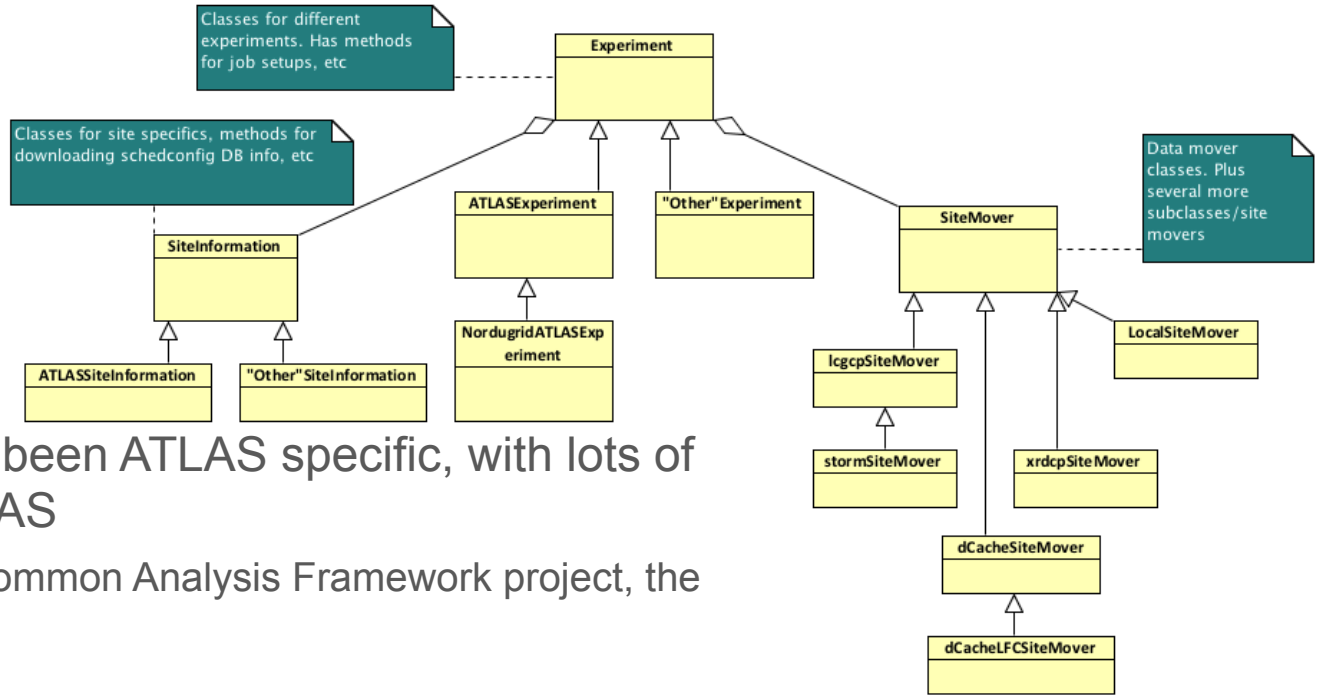
Where to find extra CPU cycles?

- Historically HEP community was not using LCF extensively
  - Early experience was not very encouraging, hardware and programming environment was not very convenient for HEP.
- Current pace of research and discovery at LHC is limited by ability of LHC computing Grid to generate Monte-Carlo events - ”Grid luminosity limit”
  - Not enough CPU power !
  - Many physics simulation requests have to wait for many month
  - Currently O(100k) CPU available to ATLAS worldwide, ¾ dedicated to MC production
- LCF are rich source of CPUs
  - Typically CPUs are weaker than on servers on the Grid, but there are many of them!
- LCF typically have good storage infrastructure
  - O(1-10PB) per installation

PanDA pilot

- PanDA is pilot based system. Pilot is what is submitted to batch queues
- PanDA pilot is an execution environment used to prepare computing element
  - Request actual payload from PanDA
  - Transfers input data from SE
  - Executes payload and monitors it during execution
  - Clean up after the payload is finished
    - Transfer output
    - Clean up, transmit logs and monitoring information
- Pilots allow for low latency job scheduling which is especially important in data analysis

Evolving Panda Pilot



- Until recently the pilot has been ATLAS specific, with lots of code only relevant for ATLAS
  - To meet the needs of the Common Analysis Framework project, the pilot is being refactored
- Experiments as plug-ins
  - Introducing new experiment specific classes, enabling better organization of the code
  - E.g. containing methods for how a job should be setup, metadata and site information handling etc, that is unique to each experiment
  - CMS experiment classes are currently being implemented
- Changes are being introduced gradually, to avoid affecting current production

PanDA Pilot for LCF

- Expanding PanDA from Grid to Leadership Class Facilities will require changes
- Each LCF is unique
  - Unique architecture and hardware
  - Specialized OS, “weak” worker nodes, limited memory per WN
  - Code cross-compilation may be required
  - Unique job submission systems
  - Unique security environment
- For BlueGene pilot submission to a worker node is typically not feasible
  - Pilot/agent per supercomputer or queue model
- This may be different for Cray platforms
- Very interested in projects by Rutgers U. group (Prof. Jha talk later today)

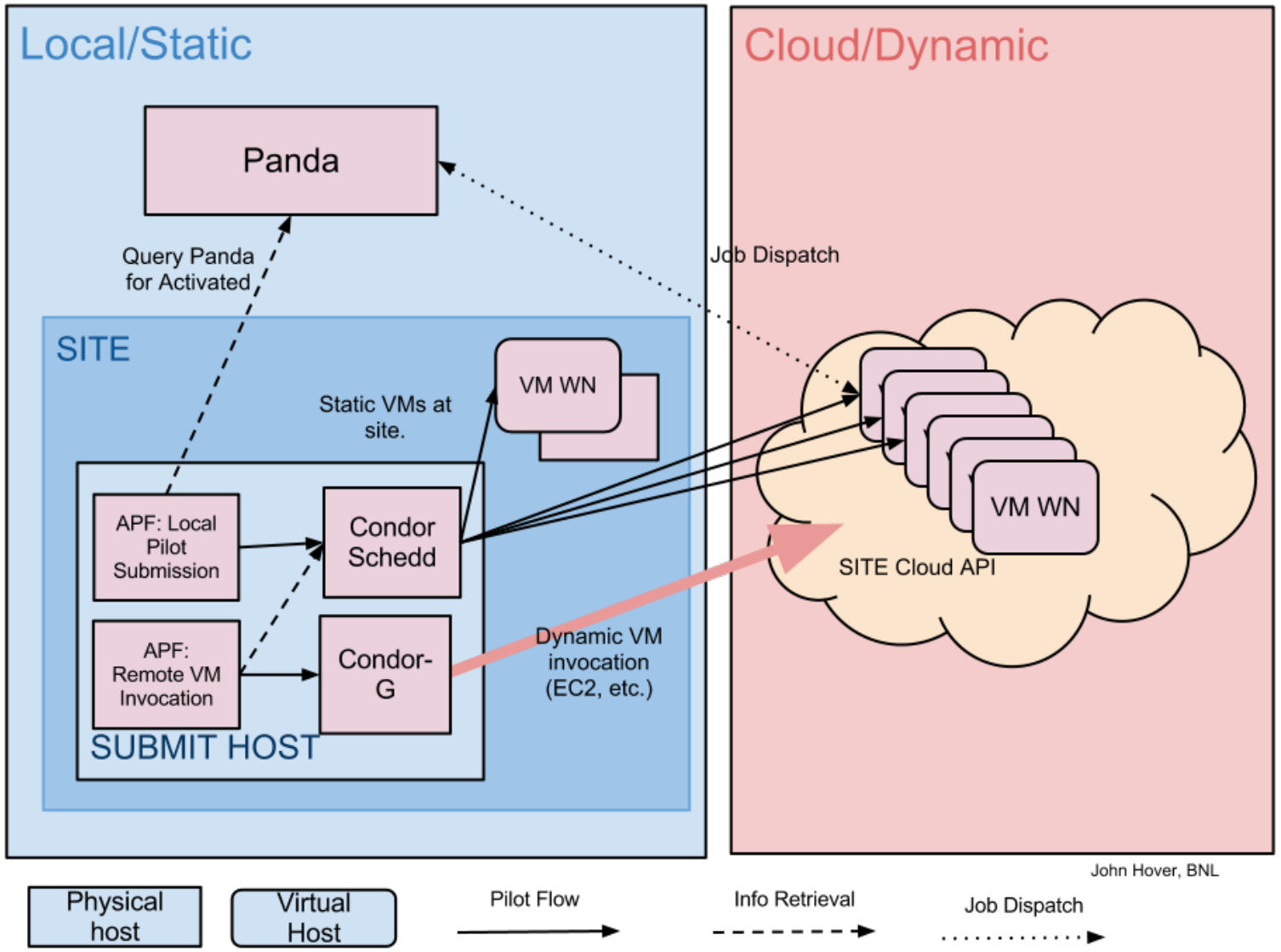
PanDA project on OLCF

- PanDA on Titan project was approved in May, 2013
- Initially thought as a precursor for Geant4 on Titan proposal
- Get experience with all relevant aspects of the platform and workload
  - job build phase – compilation, linking, etc
  - job submission mechanism
  - job output handling
  - local storage system details
  - outside transfers details
  - security environment
  - adjust monitoring model
- Develop appropriate pilot/agent model for Titan

Cloud Computing and PanDA

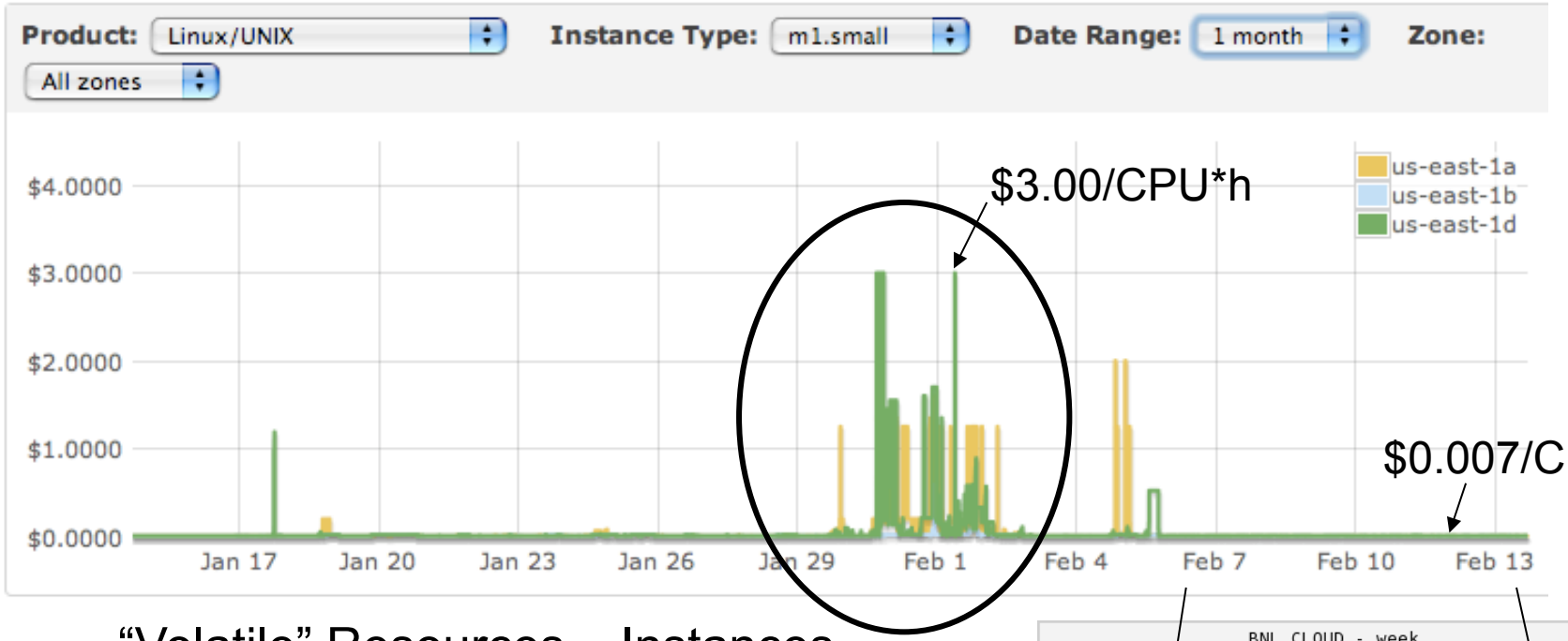
- ATLAS Distributed Computing set up a few years ago cloud computing project to exploit virtualization and clouds in PanDA
  - Utilize private and public clouds as extra computing resource
  - Mechanism to cope with peak loads on the Grid
- Experience with variety of cloud platforms
  - Amazon EC2
  - Helix Nebula (CloudSigma, T-Systems and ATOS)
  - Futuregrid (U Chicago), Synnefo cloud (U Victoria), NECTAR
  - Private clouds based on OpenStack, CloudStack, OpenNebula, etc…
  - Recent project on Google Compute Engine (GCE)

PanDA Queue on EC2



M. Ernst, J. Hover, J. Caballero

EC2 Spot Market

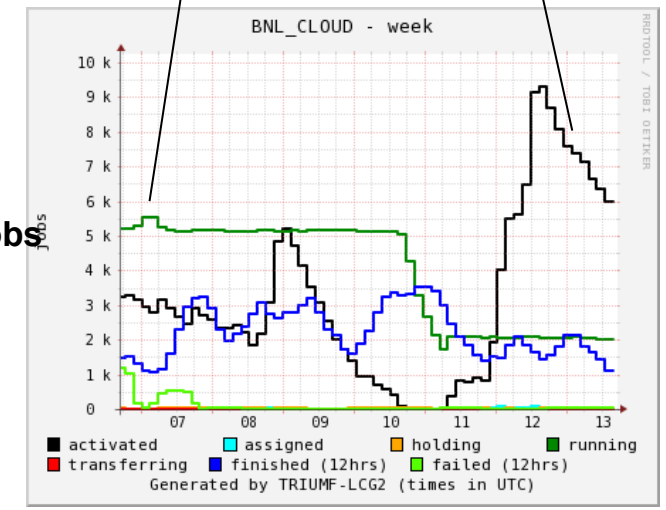


"Volatile" Resources – Instances obtained w/ lower bids are terminated

Issues w/ EC2 spot market

- User job termination w/o warning
- Even at moderately sized requests (e.g. 5k) we found it's sometimes not possible to get the number of instances one wants/needs

5k concurrent jobs ~$1,000/day



EC2 Spot Considerations

Service and Pricing

- Nodes terminated without warning (no signal).
- Partial hours are *not charged*.

Therefore, we may want to consider:

- Sub 2-hour (or even sub 1-hour) workflow generation in the ATLAS prod system.
- "Opportunistic" flag for PanDA queues, which would designate "lost heartbeat" jobs to be expected, and not considered 'failed'.
- An opportunistic flag should also result in more rapid job re-brokering within PanDA, so that jobs failed because of "lost heartbeat" get rescheduled quickly (rather than waiting for 6h).
- Per-event processing/stage-out (event server).

If we had sub-hour units of work, we could get resources for free!

ATLAS and Google Compute Engine

- We were invited to participate in GCE trial period in August 2012
- We wanted to try several things on GCE:
  - High performance analysis clusters (PROOF based and other)
  - Cloud storage and data management
  - Use of Xroot for Cloud storage aggregation and interaction with ATLAS Xroot federation
  - PanDA queue for Monte Carlo Simulations

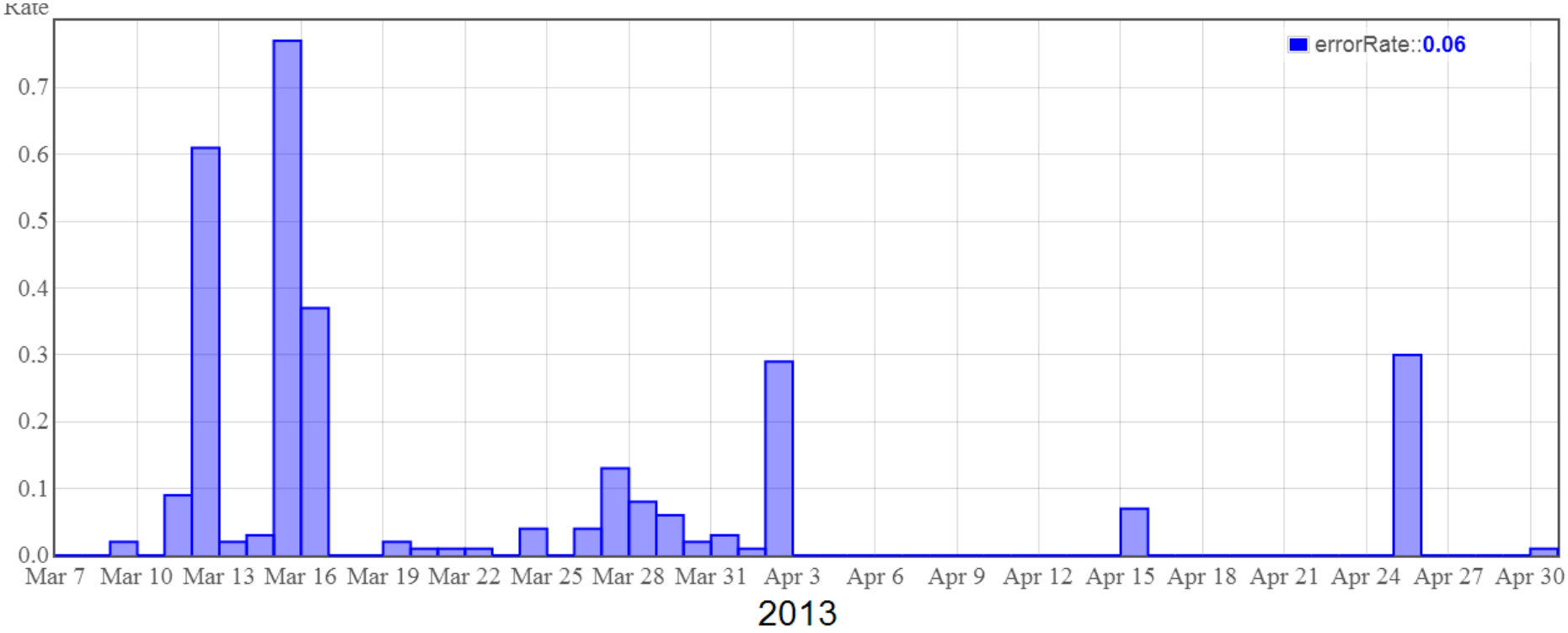
PanDA batch queue on GCE

- Google agreed to allocate additional resources for ATLAS
  - ~5M core-hours, 4000 cores for about 2 month, (original preview allocation was 1k cores)
- Resources were organized as HTCondor based PanDA queue
  - Centos 6 based custom built images, with SL5 compatibility libraries to run ATLAS software
  - Condor head nodes, proxies at BNL
  - Output exported to BNL SE
- Transparent inclusion of cloud resources into ATLAS Grid
- The idea was to test long term stability while running a cloud cluster similar in size to Tier 2 site in ATLAS
- Intended for CPU intensive Monte-Carlo simulation workloads
- Planned as a production type of run. Delivered to ATLAS as a resource and not as an R&D platform.

PanDA batch queue on GCE II

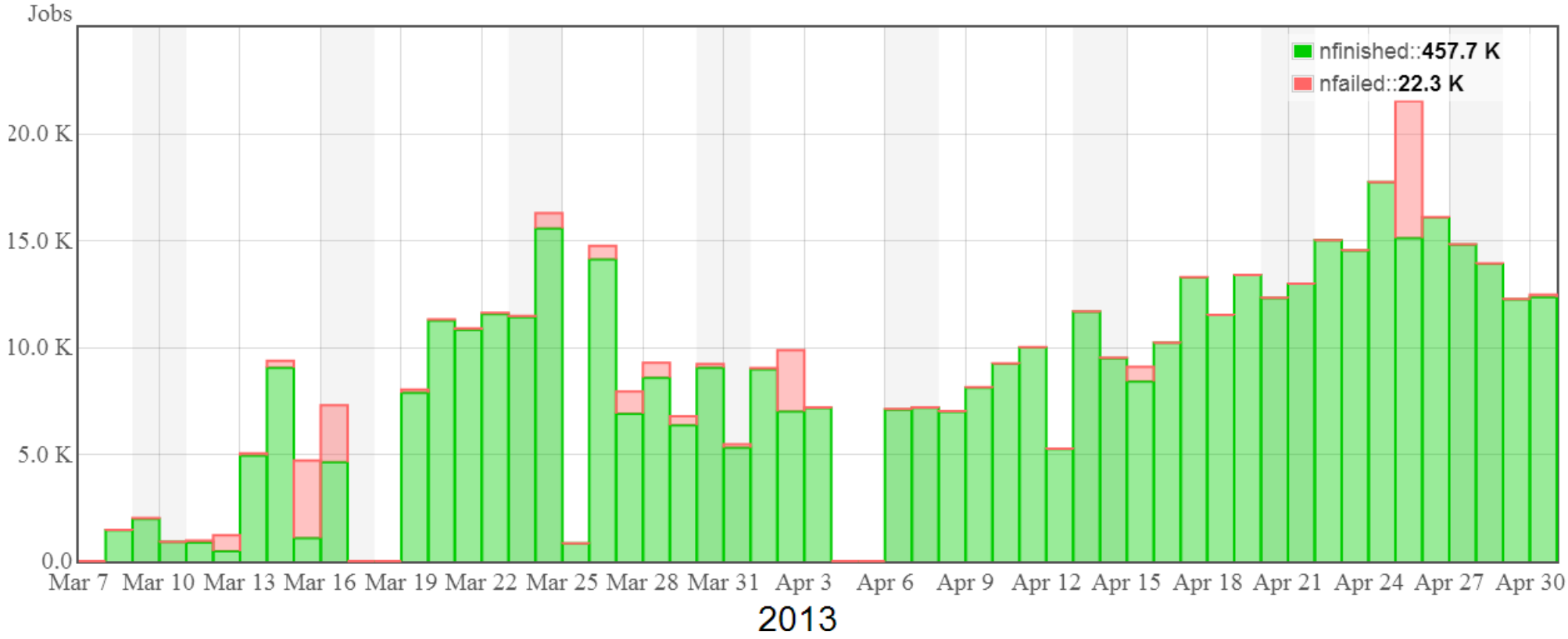
- We ran for about 8 weeks (2 weeks were planned for scaling up)
- Very stable running on the Cloud side. GCE was rock solid.
- Most problems that we had were on the ATLAS side.
- We ran computationally intensive jobs
  - Physics event generators, Fast detector simulation,, Full detector simulation
- Completed 458,000 jobs, generated and processed about 214 M events

PanDA queue on GCE. Failure Rate



- Most of the job failures occurred during start up and scale up phase – as expected
- Most of the failures were on the ATLAS side – file transfer, LFC problems, HTCondor
- No failures were due to GCE problems

Failed and Finished Jobs



- Most of the job failures occurred during start up and scale up phase – as expected
- Reached throughput of 15k jobs per day

Outlook

- Incorporation of LCF into PanDA workload management system will require significant changes in many parts of the PanDA , in particular, workload delivery on worker nodes, scope and role of pilot framework on SC, it's interaction with native SC job submission mechanisms, data management and job monitoring to name a few.
- Incorporation of IAAS resources in PanDA seems to be more transparent. Cloud management is outside of PanDA
- Participation in the Geant4 project on OLFC’s Titan will allow us to get first hand experience on this unique platform and will give us an opportunity to develop new pilot framework for Titan as a prototype of the pilot for LCF in general
- This will be an important step in the evolution of PanDA and in development of the next generation workload management system that includes LCF

The End

References

- https://twiki.cern.ch/twiki/bin/viewauth/Atlas/PanDA
- http://www.usatlas.bnl.gov/twiki/bin/view/PanDA/WebHome
- http://panda.cern.ch:25880/server/pandamon/query
- Recent Improvements in the ATLAS PanDA Pilot, P. Nilsson, CHEP 2012, United States, May 2012
- PD2P : PanDA Dynamic Data Placement for ATLAS, T. Maeno, CHEP 2012, United States, May 2012
- Evolution of the ATLAS PanDA Production and Distributed Analysis System, T. Maeno, CHEP 2012, United States, May 2012